SequenceLDhot: Detecting Hotspots

August 2006

Contents:

1 Description

The program, `sequenceLDhot`, described here is a method for detecting recombinational hotspots from population genetic data. It is described in Fearnhead (2006). It takes as input phased (i.e. haplotype) data, together with an estimate of the background recombination rate within the region (this is allowed to vary across the region). Then, `sequenceLDhot` considers a grid of putative hotspot positions, and for each putative hotspot calculates a Likelihood Ratio (LR) statistic for the presence of a hotspot (using the method of Fearnhead and Donnelly, 2002). These LR statistics can be used to give a qualitative impression of where (if anywhere) hotspots exist within the region of interest. More formally, the LR statistic values can be used to predict the position of hotspots, and a program (written in `R`) to do this is included.
For fuller details of the method, and its performance, see Fearnhead (2006).

2 Compiling the Program

`SequenceLDhot` is written in ANSI C. A makefile is provided to compile the program. Type:

```
make sequenceLDhot
```

to compile the program.

3 Input files

Either two or three input files are required by `SequenceLDhot`, and it is also possible to specify the name of the output file. The input files are:

(i) `Infile1`: sets parameters that govern the running of `SequenceLDhot`;

(ii) `Data_File`: contains the sequence data to be analysed;

(iii) `Var_Rec_File` (optional): specfies how the background recombination rate varies across the region of interest.

(Note that *all recombination rates (rho) in input and output are given per kb.*)
For input files (ii) and (iii) it is possible to directly input the output files of `PHASEv2.1` Stephens *et al.* (2001); Li and Stephens (2003); Stephens and Donnelly (2003).
To run the program type:

```
./sequenceLDhot Infile1 [-P] Data_File [-V -R Var_Rec_File] [Out_File]
```

where $[\cdots]$ means optional. The flag `-P` specifies that the `Data_File` is the (main) output file from `PHASEV2.1`. The `-V` and `-R` flags denote that a `Var_Rec_File` is to be read; with `-R` denoting that `Var_Rec_File` is the `_recom` file obtained by running `PHASEv2.1` with the `-MR` flag.

3.1 Examples

There are numerous example input files included. These can be analysed in the following ways:

```
./sequenceLDhot in1 athb3a
```

Analyse data in `athb3a` and output to default output `athb3a.sum`.

```
./sequenceLDhot in1 athb3a out
```

Analyse data in `athb3a` and output to `out.sum`.

```
./sequenceLDhot in1 -P HMIIPh_1-1-1.o
```

Analyse data in `HMIIPh_1-1-1.o` which is an output file from `PHASEv2.1`.

```
./sequenceLDhot in1 -P HMIIPh_1-1-1.o -V varrec
```

Analyse data in `HMIIPh_1-1-1.o`, assuming variable background recombination rates as specified in `varrec`.

```
./sequenceLDhot in1 -P HMIIPh_1-1-1.o -R HMMIIPh_1-1-1.o_recom
```

Analyse data in `HMIIPh_1-1-1.o`, assuming variable background recombination rates which is to be calculated from the `PHASEv2.1` output file `HMMIIPh_1-1-1.o_recom`.

3.2 Standard Input Files: an overview

The common format of the standard `Infile1` and `Data_File` files is:

- *scalar* (single value) entries are input via `text = value`, where the first character of `text` specifies the quantity whose value is being set.

- *vector* entries are input via a line of text, followed by one or more lines containing the values. Again it is the first character of the line of text which specidies the quantity that is being set.

The easiest way to input data is to edit the example input files included with the program.

3.3 Infile1

The example infile is `in1`. This file specifies (i) the length of the run; (ii) the value of the mutation rate and recombination rates; (iii) the number of driving values (see Fearnhead and Donnelly, 2001) (iv) the details of the putative hotspots to analyse.
The entries in `in1` are as follows:

```
 Number of runs = 5000
MIN number of iterations per hotspot = 100
driving values (for rho) = 2
```

These lines specify the maximum number of runs of the algorithm per putative hotspot, and the number of driving values for the recombination rate in the hotspot. (The program will stop the simulations related to a specific putative hotspot before the maximum if there is either strong evidence for it being a hotspot or of it not being a hotspot.) The values given are reasonable default values. (The number of iterations is per driving value - and the program always uses a further driving value: the background recombination rate; so in this example the total number of iterations per hotspot will be between 300 and 15,000.)

```
 background rho = 2.0
theta (per site) =0.01
```

These specify the background recombination rate (per kb); and the mutation rate (per bp). Note that if a file with variable background recombination rates is included, the background recombination rate specified here will be ignored.

```
 abs grid for hotspot likelihood
0.5 40
rel grid for hotspots likelihood
10 100
```

These lines specify the recombination rates considered within the hotspot - either in absolute values or relative to the background rate. The first two lines specify that hotspots will have recombination rates between 0.5 and 40 (per kb); the last two lines that they will have recombination rates between 10 and 100 times the background rate. If both absolute and relative values are specified, the program will take the larger of the two lower recombination rates and the smaller of the two higher recombination rates to define the range of recombination rates considered within the hotspots. So in this example, if the background recombination was 1 per kb, then it would allow for hotspot recombination rates varying between 10 and 40; if the background rate was 0.1 per kb it would allow recombination rates varying between 0.5 and 10.

```
 sub-region (number of SNPS; length (bps); frequency (bps))
7 2000 1000
#
```

The first two lines here specify the sub-regions and putative hotspots. In this case the putative hotspot have width 2000bp, and the grid considers a new hotspot every 1000bp. To calculate the LR statistic for any putative hotspots 7 SNPs will be used.
The file must end in a `#` on a new line.

3.4 Data file

Three example data files (based on data from the athb3 gene from the SeattleSNPs webiste: `http://pga.gs.washington.edu/finished_genes.html`) are included: `athb3a, athb3b, athb3c`. They are files for the same data, but in different formats.
The standard format is given in `athb3b`:

```
 Distinct =27
Genes = 94
Loci = 29
```

These specify the number of distinct haplotypes, the number of genes (or chromosomes) and the number of Loci (segregating sites/SNPs). So the data comes from 94 chromosomes, with 29 SNPs and consists of 27 distinct haplotypes.

```
 I=1 %treat data as SNPs
K = 2 %k-allele model with Haplotype Alleles by 1,2
```

The first line specifies that the data is SNPs (for sequence data include a line `B = 15000` to specify the number of bases typed) - it is reasonable to analyse sequence data as if it were SNP data. The second line specifies the number of alleles at each site (here 2). The standard input is that for a K-allele model the alleles are denoted by $1, 2, \ldots, K$ with 0 denoting an unknown allele. (There are

alternative formats: see below)

```
 Positions of loci:
326 507 1734 1913 2239 2415 2470 ....
```

These are the positions of the SNPs; in the actual file there are 29 positions (for the 29 SNPs).

```
 Haplotypes
1122 22221221212212221112 11 2 22 4
```

This is the first (of 27) Haplotypes, with its multiplicity. The first 29 numbers are the alleles at the SNP loci, (these may or may not have white space between them), then there is a space followed by the mutliplicity of the haplotype (so there are 4 of these haplotypes in the data).
(If certain sites are known to mutate quicker than others then you can add lines

```
 Variable Mutation
1 10 1 1 1 ....
```

which gives relative mutation rates at the sites - here the second site mutates tenfold quicker than the others.)
Again the file must end with a `#`.

3.4.1 Alternative formats

Two alternative formats for the data are given in `athb3a` and `athb3c`.
In `athb3a` the line `K = -4` specifies a 4-allele model where the alleles are denoted by the letters A,C,G and T (with any other character meaning an unknown allele).

In `athb3c` the line `K = -2` specifies a 2-allele model where the alleles are denoted by 0 and 1 (with any other character meaning an unknown allele).

3.4.2 Using `PHASEv2.1`

If `PHASEv2.1` was used to obtain haplotype data then using
`-P data.o`
where `data.o` is the main output file from `PHASEv2.1` will directly input the haplotype data.

3.5 Variable Background Recombination Rates

There are two ways of inputing a variable background recombination rate.

3.5.1 `-V`: Standard Input

The first way is via the `-V Var_Rec_File` flag, e.g.

```
./sequenceLDhot in1 data -V Var_Rec_File
```

In this case `-V Var_Rec_File` is just a list of recombination rates; each giving the recombination rate (per kb) for consecutive 1kb subregions of the region of interest. The first 1kb subregion starts at the position of the first SNP.
So for example

```
./sequenceLDhot in1 -P HMIIPH_1-1-1.o -V varrec
```

will analyse that `HMIIPH_1-1-1.o` data using the background rates in `varrec`. The SNPs in this data set are at positions ranging from 264 to 106905. So the first entry in `varrec` is the background rate for subregions $[264, 1263]$, the second for $[1264, 2263]$, up to the 107th which is for subregion $[106264, 107263]$. If the

file had contained more than 107 numbers, only the first 107 would have been input.

3.5.2 `-R`: Input from `PHASEv2.1`

The alternative is to get `sequenceLDhot` to calculate the rates directly from the output of `PHASEv2.1`. To do this `PHASEv2.1` must be run with the `-MR` flag, and the `_recom` file must be input using the `-R` flag, e.g.

```
./sequenceLDhot in1 -P HMIIPh_1-1-1.o -R HMIIPh_1-1-1.o_recom
```

The `HMIIPh_1-1-1.o_recom` file contains estimates of the recombination rate for between all pairs of consecutive SNPs. The background rate at a given position, x, is calculated from the estimated recombination rates in a window centered on x. The default window size includes all sequence within 50kb of x. The background rate is estimated as a specified quantile of the set of recombination rate estimates: the default is to take the median of these estimates.
The window size can be altered using the `-W` flag; and the quantile changed by using the `-Q` flag. These flags must be used immediately after the `-R` flag. For example:

```
./sequenceLDhot in1 -P HMIIPh_1-1-1.o -R HMIIPH_1-1-1.o_recom -W 20 -Q
75
```

will use a windows that contain all sequence within 20kb of x and use the upper quartile (75th %ile) for the estimate.

4 Output files

All output files are given the extension `.sum`. These files contain a list of putative hotspots considered, the estimated rate within the hotspot (which if there is no evidence for a hotspot will be the background rate), the likelihood ratio statistic for the presence of a hotspot, and a list of the position of the SNPs used to calculate the LR statistic.
For example the output of `HMIIPh_1-1-1.sum` starts:

```
Start End LR Rhohat Positions of Used SNPs
264 2264 0.000000 0.062073 264 338 805 2660 3280 3461 3942
1264 3264 0.000000 0.062318 264 338 2660 3280 3461 4467 6961
```

The first two columns of this table state the putative hotspot start and end. Next is the Likelihood Ratio (LR) value for hte presence of the hotspot (in this case 0; which shows no evidence for a hotspot). Then is the estimate of the recombination rate (per kb) with the putative hotspot – in these cases the estimate is just the background rate. The final 7 columns give the positions of the 7 SNPs used to calculate the LR value for that SNP.
Occasionally, the program may output an empty line:

```
68264 70264 0.000000 0.067632 64358 65050 71365 72432 73329 74060
69264 71264
70264 72264 0.984789 0.681521 68141 71365 72432 73329 73476 74060 7
```

This means that the estimate for the recombination rate and the LR value within the putative hotspot at position 69264–71264 is identical to that for the previous hotspot at position 68264–70264.

5 Use of output

The `R` program `HotspotSummary.R` can be used to predict hotspot positions. To load this program into `R`, run `R` within a directory that contains the file, and type

```
source(‘‘HotspotSummary.R’’)
```

To run the progrm type:

```
HotspotSummary(files,LRmax,LRmin,plot,output,ofile,title)
```

The only required argument is the first. `files` should be string that specifies the `.sum` files to analyse. If a large region has been split into smaller regions which were each analysed by `sequenceLDhot` in turn, then these can be jointly input if they have a common pattern to their name by using the `*` character in an identical way to the `ls` command in Linux. For example, if you call the output files `region_1.sum`, `region_2.sum`,. . ., `region_10.sum`, then using

```
HotspotSummary(‘‘region_*.sum’’)
```

will input all these files.
For example type `HotspotSummary(‘‘HMIIPh_1-1-1.sum’’)` to analyse the single file `HMIIPh_1-1-1.sum`, but `HotspotSummary(‘‘HMIIPh_1-1-*.sum’’)` to analyse all five files: `HMIIPh_1-1-1.sum` to `HMIIPh_1-1-5.sum`
This program will produce a list of detected hotspots and extended hotspot regions. Hotspots are only detected if the LR value is above some threshold. For any inferred hotspot there is an associated extended hotspot region which is a continguous region which all contains some evidence of a hotspot.
The optional arguments are:

- `LRmax` a numerical value that specifies the cut-off value required for inferring a hotspot.
- `LRmin` a numerical value used to define extended hotspots regions (see Fearnhead, 2006).
- `plot` either `T` or `F` depending whether you wish to have a plot of the output.
- `output` either `T` or `F` depending whether you wish to have a list of hotspots to be saved in a file.
- `ofile` the name of the output file, e.g. `ofile.txt`.
- `title` the title for the plot.

References

Fearnhead, P. (2006). SequenceLDhot: Detecting recombination hotspots. *submitted* Available from `http://www.maths.lancs.ac.uk/~fearnhea`.

Fearnhead, P. and Donnelly, P. (2001). Estimating recombination rates from population genetic data. *Genetics* **159**, 1299–1318.

Fearnhead, P. and Donnelly, P. (2002). Approximate likelihood methods for estimating local recombination rates (with discussion). *JRSS, series B* **64**, 657–680.

Li, N. and Stephens, M. (2003). Modelling LD, and identifying recombination hotspots from SNP data. *Genetics* **165**, 2213–2233.

Stephens, M. and Donnelly, P. (2003). A comparison of Bayesian methods for haplotype reconstruction from population genotype data. *American Journal of Human Genetics* **73**, 1162–1169.

Stephens, M., Smith, N. J. and Donnelly, P. (2001). A new statistical method for haplotype reconstruction from population data. *American Journal of Human Genetics* **68**, 978–989.